Chaotic Billiards

Steven Dai

Math 53, Fall 2011

Abstract

In this paper, we will investigate the simpler circular billiard and the slightly more complex (and more chaotic) stadium billiard. We will analyze the billiards for their sensitive dependence on initial conditions, measure their chaotic behavior by calculating their Lyapunov exponents and finally, some general remarks on the behaviors of both the mappings and a run-time improvement on the stadium code that was achieved through eliminating an external function call.

Introduction to Dynamical Billiard Systems

Billiards can be thought of as a mathematical model for many physical phenomena where one or more particles move in a container and collide with its walls and/or with each other. An important aspect of these billiard systems is that neither the x velocity nor the y velocity is lost or reduced in contact with the boundaries. The dynamical properties of this movement are largely determined by the shape of the walls of the container, which can vary from completely regular shapes (squares, circles, ellipses) to fully chaotic[1]. The chaotic shapes are the most intriguing, though a bit out of the scope of this paper. So, for now, we will focus on the simpler billiard within a circle and the billiard within a stadium. The stadium shape can be described as two semicircles separated by a rectangular region that can vary in size (and, as we find through testing, the chaotic behavior will also vary with the size of these straight walls).

The Circular Billiard

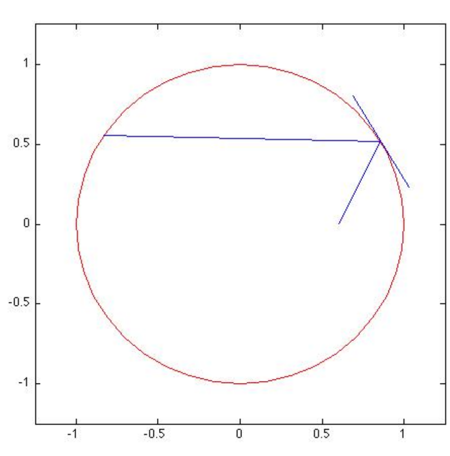

Figure 1 – Reflection of the velocity vector off the tangent vector at the boundary (in red).

The circular billiard will give us a simple introduction to the dynamics of reflection of a velocity vector in a circle and give us half of the behavior necessary to implement the stadium billiard. The circle billiard can be generated in a circle with any size radius, but for simplicity, let us assume that we are dealing with the unit circle. This means that the point of impact is always going to be at x and y such that $x^2 + y^2 = 1$. We know already that the velocities remain constant and only change direction and that change in direction is defined by the classical rule: "the angle of incidence is equal to the angle of reflection" [1]. Figure 1 shows this angle of reflection.

Because this is a continuous map of a moving particle, we can describe the motion of the billiard with the following system of equations[1]:

$$x_{t+s} = x_t + u_t s \qquad u_{t+s} = u_t$$
$$y_{t+s} = y_t + w_t s \qquad w_{t+s} = w_t$$

where (x, y) are the positions of the particle at time t and (u,w) is the velocity of vector of the vector. This holds for whenever the particle has not contacted the boundary. At that point, we must calculate the new vector as the ball moves away from the boundary. This reflection is represented by the following equation:

$$v_{new} = v_{old} - 2\langle v_{old,} n \rangle n$$

where the new velocity vector is determined by subtracting twice the dot product of the old velocity vector and normal vector at the boundary times the normal vector from the old vector. Using these rules, we are able to model the behavior of particles that move within the circle.

Notice that it is also possible to model this behavior with the following set of equations:

$$\theta_n = \theta_0 + 2n\psi_0 (\bmod 2\pi)$$
$$\psi_n = \psi_0$$

where $\theta_n \in [0, 2\pi]$ is the subsequent collision point on the circle (the position is given by $(radius \cdot \cos(\theta_n), radius \cdot \sin(\theta_n))$), θ_0 is the initial collision point on the circle (again, its position is given by $(radius \cdot \cos(\theta_0), radius \cdot \sin(\theta_0))$), and ψ_0 is the angle between the velocity and tangent vectors. Given that these equations hold, we can see that all in the circular case, all the distances between the reflection points are equal and the angle of reflection remains unchanged as the particle moves through the circle.

The Stadium Billiard

The stadium billiard is slightly harder to model, as there are several cases to handle at the boundary cases. The stadium can be pictured as the union of four curves: the lines at y = radius, y = -radius with lengths a, and two halves of a circle parameterized by the vectors $(-radius \cdot \cos(t) - a/2, -radius \cdot \sin(t))$ and $(radius \cdot \cos(t) + a/2, radius \cdot \sin(t))$ where $-\pi/2 \le t \le \pi/2$ [4]. For simplicities sake, we have assumed in these simulations that the radius of the circles is one.

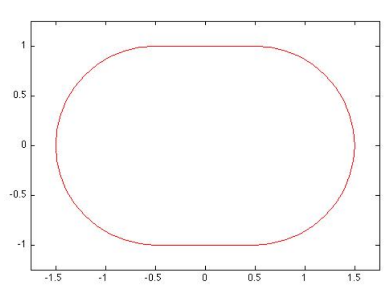

Figure 2 - Stadium Billiard with a = 1

The previous velocity and position equations still hold for the particles that move within the boundaries. But when we hit the boundaries, the reflection must either follow the reflection dictated by the circle above or the reflection of the wall. At a reflection with a straight line, the velocity vector changes in the following way:

$$(x_{vel}, y_{vel}) = (x_{vel}, -y_{vel})$$

The two types of collisions are shown below:

Sensitive Dependence on Initial Conditions

Sensitive dependence is defined by Alligood et. al as follows:

> "A point x_0 has sensitive dependence on initial conditions if there is a nonzero distance d such that some points arbitrarily near x_0 are eventually mapped at least d units from the corresponding image of x_0."

We were able to demonstrate sensitive dependence for varying initial conditions for both the circular and stadium billiards. To achieve this, we start with an initial point x_0 in both billiard systems and define another initial point y_0 a distance ε from x_0. We then iterate the billiards for a certain number of collisions until the distance between x_n and y_n is of order 1.

Let's begin with the initial point $(0, 1/\sqrt{2})$ with an initial velocity of $(1,2)$ and let $\varepsilon = 1 \times 10^{-3}$. The collisions of x_0 are shown in blue and the collisions of y_0 are shown in green. The graphs below show 30 collisions.

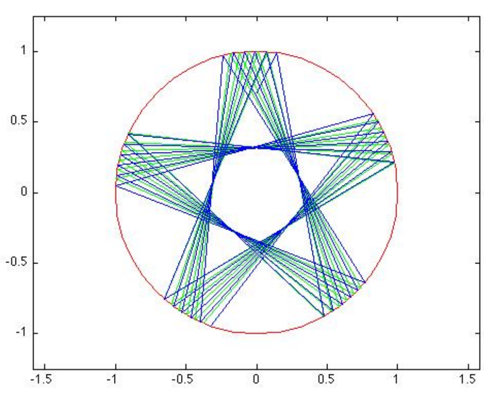

Figure 4 - Distance Between x_n and y_n vs. Collisions

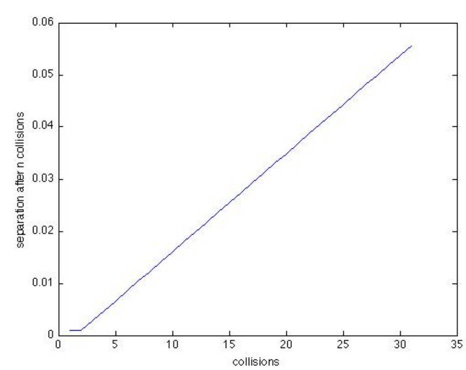

Figure 3 - Sensitive Dependence in Circular Billiard

We can see that the distance between points grows linearly as the iterations of x_n and y_n continue and the distance between x_n and y_n will soon reach order 1.

Now, let's examine the stadium billiard. We will start with the initial point (0, 0.2), initial velocity (1,0.5) and an initial separation of $\varepsilon = 1 \times 10^{-10}$. The stadium we will be using has a wall length, or a, of 2 and a circular radius of 1. Again, we will run the collision simulation for 30 collisions and the blue path represents the trajectory of x_n and the green path represents the trajectory of y_n.

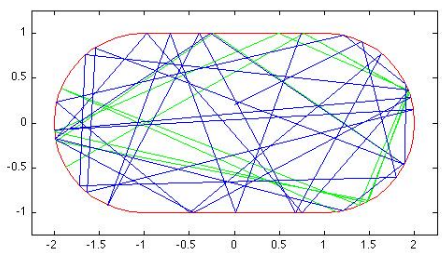

Figure 6 - Collisions in Stadium Billiard

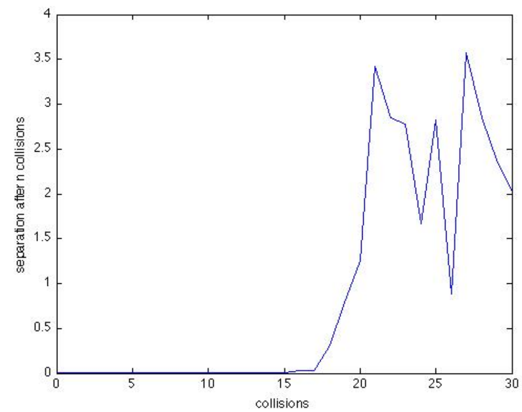

Figure 5 - Distances Between x_n and y_n

As we can see, the growth in separation starts dramatically around the 20^{th} collision of the particle against the boundary. It's already starting to look like the stadium billiard is much more chaotic than the circular billiard.

Periodic Orbits

In the circular billiard, because the reflection relies on the initial θ and ψ, as defined above, if we are able to establish a θ and a ψ such that after a certain number, let's say n, of collisions: $\theta_{n+1} = \theta_n + 2n\psi_0 = \theta_0$, then we will have a periodic orbit. That is to say, if ψ_0 is a rational multiple of π less than π, we will be able to establish a periodic orbit. Below is an example of a period-three orbit, starting at the initial point (0.5, 0) with an initial velocity of (0,1).

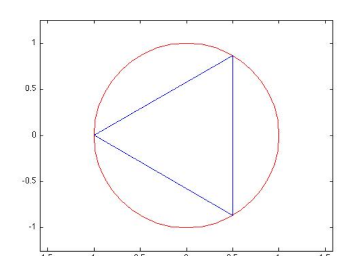

Figure 7 - Period-3 Orbit in Circular Billiard

Also, notice that if we start at the origin, any velocity vector we give this point will result in a period-2 orbit as the velocity vector will just get reflected back on its original trajectory when it hits the boundary of the circle (the velocity vector forms a 90 degree angle with the boundary tangent vector at the collision point.

Similarly, if we launch any particles originating in the stadium with a zero velocity in the y direction, we will achieve a period-2 orbit as it hits the circular ends of the stadium and reflects back on its original trajectory. If we launch any particle within the bounds of the straight-line region of the stadium with any y-velocity and a zero x-velocity, we will get a period-2 orbit that bounces between the two straight-line boundaries.

We can also achieve periodic orbits that bounce of the circular ends such as the rectangle starting from the point $(0,1/\sqrt{2})$ with an initial velocity of (1,0). The period-4 orbit is pictured below:

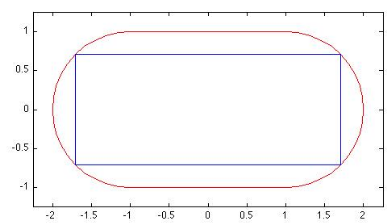

Figure 8 - Period-4 Orbit of Stadium Billiard

Lyapunov Exponents

To determine the Lyapunov exponents, we used the procedure documented in Alligood, et. al. We started with initial points separated by a small distance $\varepsilon = 1 \times 10^{-13}$ and iterated the collisions until we achieved a separation of order 1. In this case, we stopped the simulation as soon as the separation distance reached a distance of at least 0.25. When it reached this distance, we took the log of the growth divided by the initial epsilon separation and divided this entire value by the number of collisions. In symbols:

$$h = \frac{1}{collisions}\ln(\frac{dist}{\varepsilon})$$

In the circular billiard, because the growth was linear, all calculations of the Lyapunov exponent produced a number very close to zero. From Alligood, this number does not tell us much about the movement of the particle in the circular billiard but in various plots generated show much less chaotic behavior than that of the stadium.

Interestingly, the Lyapunov exponents of the stadium billiard seem to vary with the length of the straight walls of the billiard. Below, we have calculated the Lyapunov exponents of various wall lengths from 0 to 4 using the above method (Note that the stadium with wall length of 0 is just the circular billiard, so we plotted this as a number close to 0). Because of the erratic behavior of particles in this billiard, we somewhat normalized the Lyapunov exponents by starting with 30 random initial points and 30 random velocity vectors and taking the average of the calculated Lyapunov exponents of each of these various initial conditions. The plots show the maximum and the average Lyapunov exponents of each wall length.

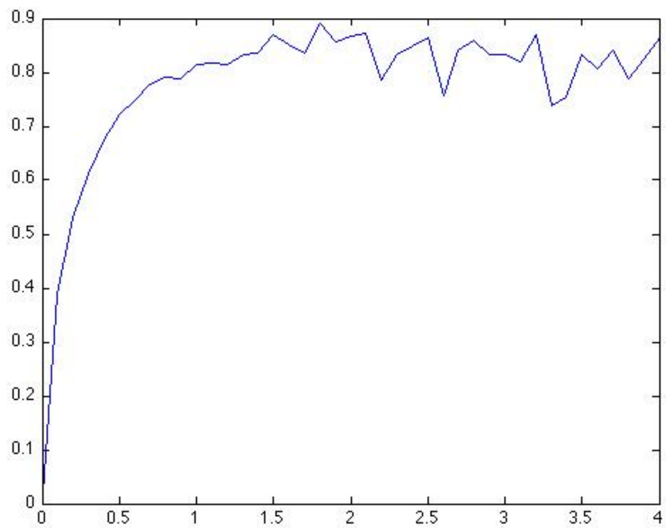

Figure 9 - Average Lyapunov Exponent of Stadium Billiard With Varying Wall Lengths

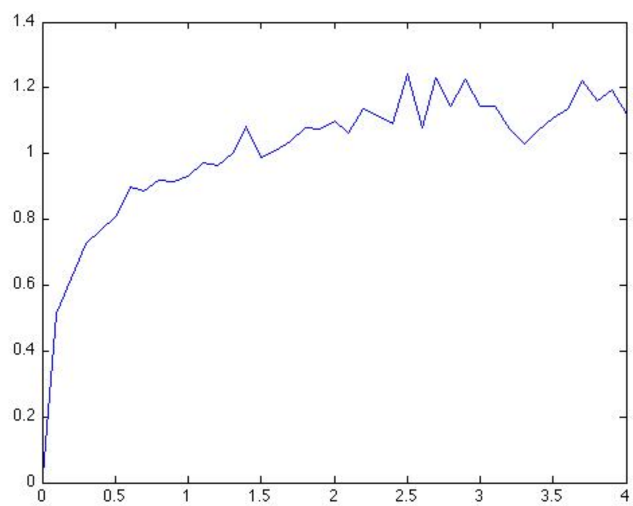

Figure 10 - Maximum Lyapunov Exponent of Stadium Billiard With Varying Wall Lengths

Notice that these Lyapunov exponents are positive, confirming our suspicion that the behavior of particles in this stadium is for the most part chaotic.

Run-time Improvements

In developing the bouncing algorithm, there were two codes developed to simulate the collisions off the boundaries. The first utilized the solve function to abstract the solving of collision points from the developer and the second used a direct mathematical approach to determine intersection points.

Here are a few examples comparing the two:

Approach 1:

```
%general intersection formula
R = sym('R');
x = sym('px + t*rx');
y = sym('py + t*ry');

…

c = x^2+y^2-R^2;
t_sol = solve(c);
x_sol = subs(x,'t',t_sol);
y_sol = subs(y,'t',t_sol);

%circle intersection
solutions = subs([x_sol y_sol],{'R' 'rx', 'ry', 'px',
'py'},{radius x_dot y_dot cir_x_0 cir_y_0});
intersection = solutions(2,:);
collision_points(:,count + 2) = [intersection(1) +
wall_length/2; intersection(2)];
```

and approach 2:

```
t = (-(2*y_point*y_dot+2*x_point*x_dot) +
sqrt((2*y_point*y_dot+2*x_point*x_dot)^2-4*(x_dot^2 +
y_dot^2)*(y_point^2 + x_point^2 -
radius^2)))/(2*(x_dot^2+y_dot^2));
```

Both of these approaches were used to solve for the intersection on the semi-circle of a velocity vector from the initial position. The y_0 was adjusted so it was the y portion of the intersection of the velocity vector with the line x = (wall length)/2. As we can see, the first approach uses a symbolic representation of the x and y direction vectors and used the solve subroutine to solve the circular equation. It also called the substitute subroutine to generate the intersection points. Contrasted with the second approach, we can see that we simply use the x and y direction vectors, represented as such:

$$x(t) = x_0 + \dot{x}t$$
$$y(t) = y_0 + \dot{y}t$$

and solved for t in the equation x. This t could then be re-applied to the system of equations to determine the intersection point on the circular end of the stadium billiard.

In the first approach, solve was used to solve for every intersection, be it with the straight line or with the circular ends while the second approach replaced these with direct mathematical equations. A quick look at the code for solve shows that there are 14 steps executed in the solve sub-routine and within the solve subroutine, it calls out to an external function as well. For a few computations, the behavior of solve is marginally different from using the mathematical equations, but as the number of calls to solve grows, the compute time grows exponentially faster than its direct equation counter-part.

To test the run-time of these approaches and to account for OS differences in load, we ran and timed 100 iterations of each approach and took the average time for these 100 iterations. For 40 collisions, the first approach took approximately 7.8 seconds while the second approach took $2.25\text{x}10^{-3}$ seconds to execute. For 20 collisions, the first approach took approximately 4.18 seconds while the second approach clocked in at $9.4\text{x}10^{-4}$ seconds.

While a decent level abstraction is nice in programming for these collisions, at the cost of a factor of a couple thousand in compute time, this approach (at least for this problem) is not worth it.

References

[1]Chernov, Nikolai, and Roberto Markarian. *Chaotic Billiards*. Mathematical Surveys and Monographs. 127. Providence: American Mathematical Society, 2006. Print.

[2]Tabachnikov, Serge. Geometry and Billiards. Providence: American Mathematical Society, 2005. Print.

[3]Alligood, Kathleen, Tim Sauer, and James Yorke. *Chaos An Introduction to Dynamical Systems*. New York: Springer, 1996. Print.

[4]Parker, Ben, and Alex Rina. "Chaotic Billiards." (2009): 1-13. Print.